

UNITED STATES PATENT AND TRADEMARK OFFICE  
**CERTIFICATE OF CORRECTION**

PATENT NO. : 6,996,565 B2  
APPLICATION NO. : 10/064965  
DATED : February 7, 2006  
INVENTOR(S) : Skufca et al.

Page 1 of 3

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 16, line 45, insert allowed claims 17-36:

-- 17. The method of claim 16, further comprising performing runtime checks prior to executing a method call including querying a security engine to determine if the method call is authorized and querying back-end adapters to determine if there are pending back-end mapped data updates for keeping cache data synchronized and updated with back-end mapped data.

18. The method of claim 1, wherein the step of creating a context definition further comprises creating a Map/Cache/Secure table.

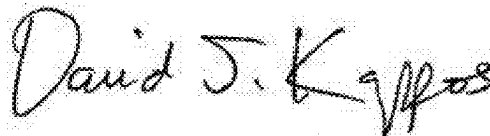
19. The method of claim 1, wherein the step of sending attribute data to clients comprises sending attribute data to client applications running on web browsers and sending attribute data to trusted Java applications running on client machines.

20. A computer-readable medium containing instructions for controlling a computer system to implement the method of claim 1.

21. A system for dynamically mapping Dynamic Multi-sourced Persisted EJB attributes to source system resources, comprising:

- means for creating a context definition for containing attributes representing collections of source system data;
- an element for specifying whether an attribute in the context definition is mapped to a field in a data source;
- means for storing the context definition in a persistent data cache within an application server;
- means for creating an instance of a Dynamic Multi-sourced Persisted EJB within the application server;
- means for applying the attributes in the context definition to the created instance of the Dynamic Multi-sourced Persisted EJB for mapping the specified attributes to source system data fields via back-end adapters;

Signed and Sealed this  
Thirtieth Day of August, 2011



David J. Kappos  
*Director of the United States Patent and Trademark Office*

means for accessing mapped source system data by the Dynamic Multi-sourced persisted EJB instance without requiring EJB compilation and deployment; and  
means for sending mapped attribute data from source systems to clients and from clients to source systems in response to client queries.

22. The system of claim 21, wherein each attribute comprises:

an element identifying a data source system table where the attribute value is located if the attribute is mapped;

an element specifying whether the attribute is cached; and

an element specifying access security requirements for the attribute.

23. The system of claim 21, further comprising means for defining a key attribute for enabling access to mapped source system data through adapters.

24. The system of claim 21, wherein each mapped attribute specified in the context definition is mapped to a single field in a data source system.

25. The system of claim 21, wherein an attribute in the context definition is designated to be mapped as a primary field in a data source and data from the primary field is written to other multiple mapped secondary fields in a data source.

26. The system of claim 21, wherein the context definition is an XML document.

27. The system of claim 21, further comprising means for storing selected source and client data in the persistent data cache.

28. The system of claim 21, wherein the means for creating an instance of a Dynamic Multi-sourced Persisted EJB comprises means for creating and accessing an instance of a Dynamic Multi-sourced Persisted EJB from an external application using generic method calls of an application programming interface selected from the group consisting of create(), find(), getAttr(), getAttrs(), getGuid(), setAttr(), setAttrs() and retrieveNewAndDeletedContexts().

29. The system of claim 28, further comprising means for performing runtime checks prior to executing a method call including means for querying a security engine to determine if the method call is authorized and means for querying back-end adapters to determine if there are pending back-end mapped data updates, for keeping cache data synchronized and updated with back-end mapped data.

30. The system of claim 21, wherein the means for creating an instance of a Dynamic Multi-sourced Persisted EJB comprises means for creating and accessing an instance of a Dynamic Multi-sourced Persisted EJB from an external application through a Session EJB Wrapper using traditional method calls of an application programming interface selected from the group consisting of create(), getAttributeName() and setAttributeName().

31. The system of claim 30, further comprising means for performing runtime checks prior to executing a method call including means for querying a security engine to determine if the method call is authorized and means for querying back-end adapters to determine if there are pending back-end mapped data updates, for keeping cache data synchronized and updated with back-end mapped data.

32. The system of claim 21, wherein a context definition comprises a Map/Cache/Secure Table.

33. The system of claim 21, wherein the means for sending attribute data to clients comprises means for sending attribute data to client applications running on web browsers and sending attribute data to trusted Java applications running on client machines.

34. A system for dynamically mapping Dynamic Multi-sourced Persisted EJB attributes to source system resources, comprising:

- an application server including contexts connected to JMS adapters;
- a data cache connected to the contexts in the application server for providing BMP data for mapping Dynamic Multi-sourced Persisted EJB attributes to back-end system data fields;
- system adapters for connecting JMS adapters to back-end systems; and
- an XML data storage device for providing context definition documents to the contexts and JMS adapters in the application server and to the system adapters.

35. The system of claim 34, wherein the contexts include Dynamic Multi-sourced Persisted EJB instances and Session EJB Wrappers mapped to source system data.

36. A system for dynamically mapping Dynamic Multi-sourced Persisted EJB attributes to source system resources, comprising:

- a context definition containing attributes representing collections of source system data;
- an attribute mapping element for specifying whether each attribute in the context definition is mapped to a field in a data source;
- the context definition being stored in a persistent data cache;
- an instance of a Dynamic Multi-sourced Persisted EJB being created;
- the attributes in the context definition being applied to the created instance of the Dynamic Multi-sourced Persisted EJB for mapping the specified attributes to source system data fields;
- mapped source system data being accessed by the Dynamic Multi-sourced Persisted EJB instance without requiring EJB compilation and deployment; and
- mapped attribute data being sent from source systems to clients and from clients to source systems in response to client queries. --